# UNITED STATES PATENT AND TRADEMARK OFFICE

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 09/599,048 | 06/21/2000 | Michael C. Murray | MS1-563US | 7795 |

| 22801　　　7590　　　11/30/2004 | EXAMINER |
|---|---|
| LEE & HAYES PLLC | STEELMAN, MARY J |

LEE & HAYES PLLC
421 W RIVERSIDE AVENUE SUITE 500
SPOKANE, WA  99201

| ART UNIT | PAPER NUMBER |
|---|---|
| 2122 | |

DATE MAILED: 11/30/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on *16 August 2004*.

2a)☐ This action is **FINAL.**     2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *1-69* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-69* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All   b)☐ Some *  c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage
application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
    Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____ .

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

## DETAILED ACTION

1.      This action is in response to Amendment B, dated 16 August 2004.

2.      Claims 1-69 are pending.

### *Information Disclosure Statement*

3.      Nonpatent literature and foreign patent references were submitted 8/23/2004, but no PTO

Form 1449 was attached.

### *Claim Rejections - 35 USC § 103*

4.      The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in
> section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are
> such that the subject matter as a whole would have been obvious at the time the invention was made to a person
> having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the
> manner in which the invention was made.

5.      Claims 1-69 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent

6,253,366 to Mutschler, III, in view of US Pre Grant Publication 2002/0032768 A1 to Voskuil.

Mutschler disclosed a (col. 1, lines 29-32) a system and method for generating a compact

Document Type Definition for data interchange...makes use of the XML entity..." ... (col. 2,

lines 37-39) "for use in a software development framework having a repository (database) and at

least two software systems..." The invention uses (col. 4, lines 25-26) "Web-based XML

standard for defining, validating and sharing document formats on the Web..." Col. 4, lines 35-

39, "The XMI specification in conjunction with the present invention will enable integration of

development tools from multiple vendors, collaboration and distribution of object-oriented

design and database schema information..."

Mutschler disclosed coded 'object models' as stored, marked-up with XML, and delivered

through a network. However, a second reference, US Patent Application 2002/0032768 A1, is

used to modify the Mutschler reference, thereby disclosing software extensions providing

functionality may also be stored in a database, converted to the XML schema, and delivered via a

network to a remote client.

**Per claim 1:**

-describing one or more software extensions using descriptions, the extensions being configured

for incorporation in a software platform executing on a client; (Col. 4, lines 21 –30 and 48-60,

"...combines the benefits of the Web-based XML standard for defining, validating and sharing

document formats on the Web...." and "...generate Document Type Definitions...for the

Extensible Markup Language...")

Mutschler disclosed (Col. 2, lines 19-22), "Another object of the present invention is to provide a

method and system that allows developers of distributed systems the ability to share object

models and other meta-data over a network, including the Internet (enable network delivery)."

Col. 2, lines 27-31, "A feature of the present invention is the use of entity objects to encapsulate

properties and behaviors of each class object thereby making the document type definition

(DTD) more compact and giving a clearer picture of the relationships in the meta-model being

captured."

Mutschler failed to disclose "software delivery over a network".

However, Voskuil disclosed:

-delivering the descriptions of the one or more extensions to the client via a network, the

descriptions being configured for use in downloading the software extensions via the network.

([0014], "The client application can have the ability to be extended by external or third party

software components or modules...a client application extension that can be transmitted to and

installed on the end user computer...", [0029], "...instructions can be incorporated in an XML

data structure that is transmitted from the server...")

-said acts of describing and delivering being configured to enable software to be delivered over

the network.

([0029], "The client application supports an extension interface...allow the client application to

be extended to permit additional functionality.  The extension interface permits extension

modules (e.g. plug-ins or Active-X controls) to access the file system to create, read and write to

files on the remote PC...")

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention to modify Mutschler's invention to include a repository of software extensions to be

transmitted to a client via the network, because Mutschler's code for object models or code for

software extensions, as disclosed by Voskuil, is merely code. Both are stored in a repository /

database and delivered to a remote location using XML and a description of the content, in the

form of a DTD document. Both references support (Mutschler, col. 2, lines 4-5), "a need for

automatically generating format descriptions to expedite interchange..." using a common

language, (Mutschler, col. 4, lines 36-39), to "enable integration of development tools from

multiple vendors, collaboration and distribution of object-oriented design..."


**Per claim 2:** (Col. 2, line 22.)

**Per claim 3:** (Figs. 3A, 3B, & 3C and col. 7, lines 20-25.)

**Per claim 4:** (Figs. 3A, 3B, and 3C and col. 4, lines 21-28.)

**Per claim 5:** (Col. 2, line 22 and col. 4, lines 21-28.)

**Per claim 6:** (Col. 4, lines 31-39.)

**Per claims 7 & 8:** -wherein causing said computer system to describe one or more extensions

and deliver XML descriptions enables software to be delivered over the Internet. (Col. 4, lines

31-39, "The XMI specification provides application developers with a common language for

specifying transfer syntax for development language that allow visualizing, constructing and

documenting of distributed objects and business models (software)." Col. 2, lines 27-31, "A

feature of the present invention is the use of entity objects to encapsulate properties and

behaviors of each class object thereby making the document type definition (DTD) (describe

extensions) more compact and giving a clearer picture of the relationships in the meta-model

being captured." Col. 2, lines 21-22, "...allows developers of distributed systems the ability to

share object models and other meta-data over a network, including the Internet. (deliver over

Internet)" Col. 5, lines 16-20, "...the repository is a specialized, extensible object-oriented

database application that adds value to a database system, which allows customization of a

particular domain (such as application development)." A repository of object oriented object

models is used as a software development framework.)


**Per claim 9:** (Col. 4, lines 31-35.)

**Per claim 10:** (Col. 3, line 60 – col. 4, line 9.)

**Per claim 11:** (Col. 4, lines 1-10.)

**Per claim 12:** (Col. 4, lines 11-18.)

**Per claim 13:**

See rejection of limitations addressed in claims 9-12 above.

**Per claim 14:** (Col. 4, lines 21-28 and 29-39.)

**Per claim 15:** (Col. 4, lines 48-56.)

**Per claim 16:** (Col. 4, lines 56-60.)

**Per claim 17:**

-computer readable media...

(Col. 16, lines 30-35, "...program code embodied in tangible media...")


-describe one or more software extensions using extensible markup language (XML), the

extensions being configured for incorporation in a software platform comprising a single

application program, the single application program having multiple different functionalities that

can enable a user to accomplish multiple different tasks;

(Col. 4, lines 21 –30 and 48-60, "...combines the benefits of the Web-based XML standard for defining, validating and sharing document formats on the Web...." and "...generate Document Type Definitions...for the Extensible Markup Language...")

Mutschler disclosed (Col. 2, lines 19-22), "Another object of the present invention is to provide a method and system that allows developers of distributed systems the ability to share object models and other meta-data over a network, including the Internet (enable network delivery)." Col. 2, lines 27-31, "A feature of the present invention is the use of entity objects to encapsulate properties and behaviors of each class object thereby making the document type definition (DTD) more compact and giving a clearer picture of the relationships in the meta-model being captured."

Mutschler failed to disclose "software delivery over a network".

However, Voskuil disclosed:

-deliver XML descriptions of the one or more extensions to the client via the Internet, the descriptions being configured for use in downloading the software extensions via the Internet; ([0014], "The client application can have the ability to be extended by external or third party software components or modules...a client application extension that can be transmitted to and installed on the end user computer...", [0029], "...instructions can be incorporated in an XML data structure that is transmitted from the server...")

-wherein causing said computer system to describe one or more extensions and deliver XML

descriptions enables software to be delivered over the Internet.

([0029], "The client application supports an extension interface...allow the client application to

be extended to permit additional functionality. The extension interface permits extension

modules (e.g. plug-ins or Active-X controls) to access the file system to create, read and write to

files on the remote PC...")


Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention to modify Mutschler's invention to include a repository of software extensions to be

transmitted to a client via the network, because Mutschler's code for object models or code for

software extensions, as disclosed by Voskuil, is merely code. Both are stored in a repository /

database and delivered to a remote location using XML and a description of the content, in the

form of a DTD document. Both references support (Mutschler, col. 2, lines 4-5), "a need for

automatically generating format descriptions to expedite interchange..." using a common

language, (Mutschler, col. 4, lines 36-39), to "enable integration of development tools from

multiple vendors, collaboration and distribution of object-oriented design..."


**Per claim 18:**

-describing one or more software extensions using one or more descriptive files, the extensions

being configured for incorporation in a software program executing on a client; (Col. 4, lines 21-

30 and 48-60 and col. 6, lines 13-16. Col. 4, lines 54-60, "The DTD also declares all the external

entities referenced within the document and the notations that can be used. Stated otherwise, an

XML DTD (descriptive file) provides a means by which an XML processor can validate the

syntax and some of the semantics of an XML document (extension file). An XMI DTD specifies

the particular elements allowed in an XMI document." Additionally, note col. 2, lines 19-22,

"Another object of the present invention is to provide a method and system that allows

developers of distributed systems the ability to share object models and other meta-data over a

network, including the Internet." The XMI document may be an object model. Col. 3, lines 60-

64, "An object is an abstract representation of a real-world concept or thing...An object has

features, which can be either an operation or a property. An operation defines an action that an

object can perform (program functionality)...")

-storing the descriptive files and associated extension files in a network-accessible location;

(Fig. 2, DTD.)

-delivering the descriptive files and the associated extension files of the one or more extensions

to the client via a network; (Fig. 1.)

Mutschler fails to provide program functionality. However Voskuil disclosed:

-associating the one or more descriptive files with one or more associated extension files that are

useable to provide a program functionality;

([0015], "Each module can include one or more functions that can be used to

access...search...modify...or create...")

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention to modify Mutschler's invention to include details associating descriptive files with

associated extension files to provide program functionality, because both inventions provide for

code stored in a repository / database and delivered to a remote location using XML using a

description of the content, in the form of a DTD document. Both references support (Mutschler,

col. 2, lines 4-5), "a need for automatically generating format descriptions to expedite

interchange..." using a common language, (Mutschler, col. 4, lines 36-39), to "enable integration

of development tools from multiple vendors, collaboration and distribution of object-oriented

design..." The delivery of XML code and an associated description document are well known in

the art.

**Per claim 19:** (Col. 4, lines 48-56.)

**Per claim 20:**

See rejection of limitations as addressed in claim 6 above.

**Per claim 21:**

See rejection of limitations as addressed in claim 7 above.

**Per claim 22:**

See rejection of limitations as addressed in claim 9 above.

**Per claim 23:**

See rejection of limitations as addressed in claim 9 above.

**Per claim 24:**

See rejection of limitations as addressed in claim 10 above.

**Per claim 25:**

See rejection of limitations as addressed in claim 11 above.

**Per claim 26:**

See rejection of limitations as addressed in claim 12 above.

**Per claim 27:**

See rejection of limitations as addressed in claims 9-12 above.

**Per claim 28:**

See rejection of limitations as addressed in claim 18 above.

**Per claim 29:**

Mutschler disclosed:

-storing one or more definition files (EDFs); (Figs. 1 and 2, DTD.)


Mutschler failed to disclose:

-extension definition files (EDFs) that describe a logical attachment to a software application...

 -storing one or more extension files that correspond to the one or more EDFs and extend the

software application program;

-delivering, via the network, at least one EDF to a client; delivering, via a network, at least one

extension file that corresponds to the at least one EDF to a client.

-both of said acts of storing and both of said acts of delivering enabling software to be delivered

over the network.


However Voskuil disclosed:

[0014], "The client application can have the ability to be extended by external or third party software components or modules...a client application extension that can be transmitted to and installed on the end user computer...", [0015], "Each module can include one or more functions that can be used to access files on the end user computer, search...modify...or create...", "modules can be instructed...to execute one or more of the library of functions by a list of instructions (extension definition files that describe a logical attachment to a software application) received from the...server (delivered via the network)."

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention to modify Mutschler's invention to include a repository of software extensions to be transmitted to a client via the network, because Mutschler's code for object models or code for software extensions, as disclosed by Voskuil, is merely code. Both are stored in a repository / database and delivered to a remote location using XML and a description of the content, in the form of a DTD document or instructions. Both references support (Mutschler, col. 2, lines 4-5), "a need for automatically generating format descriptions to expedite interchange..." using a common language, (Mutschler, col. 4, lines 36-39), to "enable integration of development tools from multiple vendors, collaboration and distribution of object-oriented design..."

**Per claim 30:**

See rejection of limitations as addressed in claim 2 above.

**Per claim 31:** (Col. 2, line 22.)

**Per claim 32:** (Col. 4, lines 65-66.)

**Per claim 33:** (Col. 4, lines 48-60.)

**Per claims 34:** (Col. 4, lines 51-54.)

**Per claim 35:**

See rejection of limitations as addressed in claim 9 above.

**Per claim 36:**

See rejection of limitations as addressed in claim 10 above.

**Per claim 37:**

See rejection of limitations as addressed in claim 11 above.

**Per claim 38:**

See rejection of limitations as addressed in claim 12 above.

**Per claim 39:**

See rejection of limitations as addressed in claim 29 above.


**Per claim 40:**

Mutschler disclosed:

-a first sub-structure indicative of a software extension that is to be incorporated...;

(Col. 2, lines 19-26, "...developers of distributed systems the ability to share...as streams or files

with a standard format based on XML."


As defined at google.com, XML: eXtensible Mark-up Language, a specification developed by

the W3C. XML is a pared-down version of Standard Generalised Mark-Up Language, designed

especially for Web documents. It allows designers to create their own customised tags, enabling

the definition, transmission, validation, and interpretation of data between applications and

between organizations.

Therefore, it is inherent that the software is extendable.

-one or more second sub-structures associated with the first sub-structure and indicating feature

types that are added by the extension to the application program;

Feature Types (as found in Specification page 14, lines 12-19): EDFs can extend the extension

mechanism...by creating their own tags...extensions can be extended...EDFs can also have one

or more predefined tags...predefined XML tags for user interface elements can include tags for

feature types such as tool bars, accelerators, menu items, and themes...utilized in the single

navigable window application.

The definition of XML allows for extensions to be extended, i.e., a 'feature type' as defined in

the Specification may have a tag created, and may be further extended.

-one or more third sub-structures associated with the one or more second sub-structures and

indicating features of an associated feature type that are added by the extension.

(Col. 4, lines 1-8 and 21-39.  Col. 6, lines 35-36, "There are various methods by which the DTD

(EDF) generator can produce the DTD."  Mutschler disclosed (Col. 6, lines 50-57), "The method

of the present invention allows for grouping of the parts of an object into XML entity definitions

(customize extensions)...since the Attributes, References and Compositions of an object are

defined in only one place, modification (customized extensions) is greatly simplified."

Mutschler failed to disclose "software application program". However Voskuil disclosed [0029],

"client application to be extended to permit additional functionality"...using plug-ins or Active-

X controls...incorporated in an XML data structure transmitted from the server. Thus Voskuil

disclosed a "software application program"

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention, to modify combined invention to include details that provide for a data structure to

comprise of a first sub-structure indicative of a software extension, one or more second sub-

structures associated with the first sub-structure, indicating feature types (extended extensions

with tags) that are added, and one or more third sub-structures associated with the one or more

second sub-structures, indicating features of an associated feature type that are added by the

extension, because XML is an object based hierarchical programming construct, that by the very

definition is extensible, therefore any software extension may be further extended to provide

additional functionality.

**Per claims 41, 42 and 43:**

As related to substructures being children (derived / extended) of a parent structure:

As defined at google.com, XML: eXtensible Mark-up Language, a specification developed by

the W3C. XML is a pared-down version of Standard Generalised Mark-Up Language, designed

especially for Web documents. It allows designers to create their own customised tags, enabling

the definition, transmission, validation, and interpretation of data between applications and

between organizations.


The definition of XML allows for extensions to be extended, i.e., a 'feature type' as defined in

the Specification may have a tag created, and may be further extended.


Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention, to modify Mutschler's invention to include details that provide for a data structure to

comprise of a first sub-structure indicative of a software extension, one or more second sub-

structures associated with the first sub-structure, indicating feature types (extended extensions

with tags) that are added, and one or more third sub-structures associated with the one or more

second sub-structures, indicating features of an associated feature type that are added by the

extension, because XML is an object based hierarchical programming construct, that by the very

definition is extensible, therefore any software extension may be further extended to provide

additional functionality.


**Per claim 44:**

As related to substructures being children (derived / extended) of a parent structure:

As defined at google.com, XML: eXtensible Mark-up Language, a specification developed by

the W3C. XML is a pared-down version of Standard Generalised Mark-Up Language, designed

especially for Web documents. It allows designers to create their own customised tags, enabling

the definition, transmission, validation, and interpretation of data between applications and

between organizations.

The definition of XML allows for extensions to be extended, i.e., a 'feature type' as defined in

the Specification may have a tag created, and may be further extended.

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention, to modify Mutschler's invention to include details that provide for a data structure to

comprise of a first sub-structure indicative of a software extension, one or more second sub-

structures associated with the first sub-structure, indicating feature types (extended extensions

with tags) that are added, and one or more third sub-structures associated with the one or more

second sub-structures, indicating features of an associated feature type that are added by the

extension, because XML is an object based hierarchical programming construct, that by the very

definition is extensible, therefore any software extension may be further extended to provide

additional functionality.

**Per claim 45:**

Voskuil disclosed:

-feature types comprise one or more of the following feature types: user interface elements;

behaviors, components, or objects; store elements; and user-defined elements;

([0029], "...extended to permit additional functionality (behaviors)..."

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention, to modify Mutschler's invention to include details that provide for extending software

to permit additional behaviors, because although Mutschler's invention involved model code,

software application code is handled in the same manner, as disclosed by Voskuil, with updates

or enhanced functionality, available through a network, provided using the XML schema.

**Per claims 46 and 47:**

Voskuil disclosed:

([0014], "The client application can have the ability to be extended by external or third party

software..."

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention, to modify Mutschler's invention to include details that provide for extending software

to permit additional behaviors, because although Mutschler's invention involved model code,

software application code is handled in the same manner, as disclosed by Voskuil, with updates

or enhanced functionality, available through a network, provided using the XML schema.

**Per claim 48:**

Mutschler disclosed (Col. 2, lines 19-22), "Another object of the present invention is to provide a

method and system that allows developers of distributed systems the ability to share object

models and other meta-data over a network, including the Internet (enable network delivery)."

Col. 2, lines 27-31, "A feature of the present invention is the use of entity objects to encapsulate

properties and behaviors of each class object thereby making the document type definition

(DTD) more compact and giving a clearer picture of the relationships in the meta-model being

captured."

Mutschler failed to disclose "software delivery over a network".

However, Voskuil disclosed:

-delivering the descriptions of the one or more extensions to the client via a network, the

descriptions being configured for use in downloading the software extensions via the network.

([0014], "The client application can have the ability to be extended by external or third party

software components or modules...a client application extension that can be transmitted to and

installed on the end user computer...", [0029], "...instructions can be incorporated in an XML

data structure that is transmitted from the server...")

-said acts of describing and delivering being configured to enable software to be delivered over

the network.

([0029], "The client application supports an extension interface...allow the client application to

be extended to permit additional functionality. The extension interface permits extension

modules (e.g. plug-ins or Active-X controls) to access the file system to create, read and write to

files on the remote PC...")


Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention to modify Mutschler's invention to include a repository of software extensions to be

transmitted to a client via the network, because Mutschler's code for object models or code for

software extensions, as disclosed by Voskuil, is merely code. Both are stored in a repository /

database and delivered to a remote location using XML and a description of the content, in the

form of a DTD document. Both references support (Mutschler, col. 2, lines 4-5), "a need for

automatically generating format descriptions to expedite interchange..." using a common

language, (Mutschler, col. 4, lines 36-39), to "enable integration of development tools from

multiple vendors, collaboration and distribution of object-oriented design..."


-navigating to a network site...;

-downloading...from the network site...comprising multiple different functionalities that can

assist a user in accomplishing different tasks...configured to be extended with software

extensions that are deliverable via a network and are described by at least one network-

deliverable file.

(Col. 4, lines 24-39 and col. 5, lines 18-19. Col. 1, lines 49-52, "Repository models typically

contain classes, data types and messages. As more and more complex models are being built, the

need arises for a method and system to transfer data in a model from place to place..." Col. 2,

lines 19-22, "Another object of the present invention is to provide a method and system that

allows developers of distributed systems the ability to share object models and other meta-data

over a network, including the Internet." Col. 2, lines 27-31, "A feature of the present invention

is the use of entity objects to encapsulate properties and behaviors of each class object thereby

making the document type definition (DTD) more compact and giving a clearer picture of the

relationships in the meta-model being captured.")

Mutschler failed to disclose "software delivery over a network".

However, Voskuil disclosed:

([0014], "The client application can have the ability to be extended by external or third party

software components or modules (software delivery)…a client application extension that can be

transmitted to and installed on the end user computer…", [0029], "…instructions can be

incorporated in an XML data structure that is transmitted from the server (download to

client)…", [0029], "The client application supports an extension interface…allow the client

application to be extended to permit additional functionality. The extension interface permits

extension modules (e.g. plug-ins or Active-X controls) to access the file system to create, read

and write to files on the remote PC…")

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention to modify Mutschler's invention to include a repository of software extensions to be

transmitted to a client via the network, because Mutschler's code for object models or code for

software extensions, as disclosed by Voskuil, is merely code. Both are stored in a repository /

database and delivered to a remote location using XML and a description of the content, in the

form of a DTD document. Both references support (Mutschler, col. 2, lines 4-5), "a need for

automatically generating format descriptions to expedite interchange..." using a common

language, (Mutschler, col. 4, lines 36-39), to "enable integration of development tools from

multiple vendors, collaboration and distribution of object-oriented design..."


**Per claim 49:**

See rejection of limitations as addressed in claim 9 above.

**Per claim 50:**

Voskuil disclosed:

-extending the software application program by adding at least one extension

([0014], "The client application can have the ability to be extended by external or third party

software components or modules (software delivery)...", [0029], "The client application

supports an extension interface...allow the client application to be extended to permit additional

functionality. The extension interface permits extension modules (e.g. plug-ins or Active-X

controls) to access the file system to create, read and write to files on the remote PC...")


Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention to modify Mutschler's invention to include a repository of software extensions to be

transmitted to a client via the network, because Mutschler's code for object models or code for

software extensions, as disclosed by Voskuil, is merely code. Both are stored in a repository /

database and delivered to a remote location using XML and a description of the content, in the

form of a DTD document. Both references support (Mutschler, col. 2, lines 4-5), "a need for automatically generating format descriptions to expedite interchange..." using a common language, (Mutschler, col. 4, lines 36-39), to "enable integration of development tools from multiple vendors, collaboration and distribution of object-oriented design..." Using XML allows the program to be extended.

**Per claim 51:**

-using a link to navigate to a different network site that hosts one or more XML files that describe the extension, and extension files that are used to implement the extension;

-downloading the one or more XML files and the extension files to a client.

(Col. 4, lines 24-26.)

**Per claim 52:** (Col. 4, lines 29-35 and col. 6, lines 20-23.)

**Per claim 53:** (Col. 4, lines 16-20 and line 65 – col. 5, line 4.)

**Per claim 54:**

Mutschler disclosed:

-navigating to a network site...;

-downloading...from the network site...comprising multiple different functionalities that can assist a user in accomplishing different tasks...configured to be extended with software extensions that are deliverable via a network and are described by at least one network-deliverable file.

(Col. 4, lines 24-39 and col. 5, lines 18-19. Col. 1, lines 49-52, "Repository models typically

contain classes, data types and messages. As more and more complex models are being built, the

need arises for a method and system to transfer data in a model from place to place..." Col. 2,

lines 19-22, "Another object of the present invention is to provide a method and system that

allows developers of distributed systems the ability to share object models and other meta-data

over a network, including the Internet." Col. 2, lines 27-31, "A feature of the present invention

is the use of entity objects to encapsulate properties and behaviors of each class object thereby

making the document type definition (DTD) more compact and giving a clearer picture of the

relationships in the meta-model being captured.")


Mutschler failed to disclose "software delivery over a network".

However, Voskuil disclosed:

([0014], "The client application can have the ability to be extended by external or third party

software components or modules (software delivery)...a client application extension that can be

transmitted to and installed on the end user computer...", [0029], "...instructions can be

incorporated in an XML data structure that is transmitted from the server (download to

client)...", [0029], "The client application supports an extension interface...allow the client

application to be extended to permit additional functionality. The extension interface permits

extension modules (e.g. plug-ins or Active-X controls) to access the file system to create, read

and write to files on the remote PC...")

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention to modify Mutschler's invention to include a repository of software extensions to be

transmitted to a client via the network, because Mutschler's code for object models or code for

software extensions, as disclosed by Voskuil, is merely code. Both are stored in a repository /

database and delivered to a remote location using XML and a description of the content, in the

form of a DTD document. Both references support (Mutschler, col. 2, lines 4-5), "a need for

automatically generating format descriptions to expedite interchange..." using a common

language, (Mutschler, col. 4, lines 36-39), to "enable integration of development tools from

multiple vendors, collaboration and distribution of object-oriented design..."


**Per claims 55:**

Mutschler disclosed:

-accessing a web site through which one or more extensions can be obtained and delivered;

(Col. 6, lines 11-16.)

-receiving at least one file that describes at least one extension using a hierarchical language that

describes the extension's logical attachment to a software application program;

-receiving one or more extension files;

-installing the one or more extension files based, at least in part, on the description contained in

said at least one file. (Col. 4, lines 21-39 and col. 6, lines 29-49.)


Mutschler failed to disclose "software delivery over a network".

However, Voskuil disclosed:

-software delivery of software extensions over a network

([0014], "The client application can have the ability to be extended (software extensions) by external or third party software components or modules...a client application extension that can be transmitted to and installed on the end user computer...", [0029], "...instructions can be incorporated in an XML data structure that is transmitted from the server...")

([0029], "The client application supports an extension interface...allow the client application to be extended to permit additional functionality. The extension interface permits extension modules (e.g. plug-ins or Active-X controls) to access the file system to create, read and write to files on the remote PC...")

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the invention to modify Mutschler's invention to include a repository of software extensions to be transmitted to a client via the network, because Mutschler's code for object models or code for software extensions, as disclosed by Voskuil, is merely code. Both are stored in a repository / database and delivered to a remote location using XML and a description of the content, in the form of a DTD document. Both references support (Mutschler, col. 2, lines 4-5), "a need for automatically generating format descriptions to expedite interchange..." using a common language, (Mutschler, col. 4, lines 36-39), to "enable integration of development tools from multiple vendors, collaboration and distribution of object-oriented design..."

**Per claims 56 and 57:**

See rejection of limitations as addressed in claims 3 and 4 above.

**Per claim 58:** (Col. 6, lines 33-35.)

**Per claim 59:** (Col. 6, lines 50-51.)

**Per claims 60 and 61:** (Col. 5, lines 21-23.)

**Per claim 62:**

-computer readable media... (Col. 16, lines 30-34)

See rejection of limitations as addressed in claim 55 above.

**Per claim 63:**

Mutschler disclosed:

-describing one or more extensions using one or more extensible markup language (XML) files,

-storing the XML files and associated extension files in a network-accessible location. (Col. 4, lines 21-39.)

-said acts of describing and associating being configured to provide delivery over the network. (Col. 2, lines 20-22.)


Mutschler failed to disclose:

-extensions configured for incorporation in a software program; -associating the one or more XML files with one or more associated extension files that are useable to provide a program functionality; the extensions being configured for incorporation in a software program executing on a client; to provide for software delivery over the network.


However, Voskuil disclosed:

([0014], "The client application can have the ability to be extended by external or third party

software components or modules (extensions configured for incorporation in a software

program)...a client application extension that can be transmitted to (delivery over the network)

and installed on the end user computer...", [0029], "...instructions can be incorporated in an

XML data structure that is transmitted from the server...", ([0029], "The client application

supports an extension interface...allow the client application to be extended to permit additional

functionality (provide program functionality). The extension interface permits extension

modules (e.g. plug-ins or Active-X controls) to access the file system to create, read and write to

files on the remote PC...")


Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention to modify Mutschler's invention to include a repository of software extensions to be

transmitted to a client via the network, because Mutschler's code for object models or code for

software extensions, as disclosed by Voskuil, is merely code. Both are stored in a repository /

database and delivered to a remote location using XML and a description of the content, in the

form of a DTD document. Both references support (Mutschler, col. 2, lines 4-5), "a need for

automatically generating format descriptions to expedite interchange..." using a common

language, (Mutschler, col. 4, lines 36-39), to "enable integration of development tools from

multiple vendors, collaboration and distribution of object-oriented design..."


**Per claim 64:**

Mutschler disclosed:

-one or more software extension files configured, the software extension files being configured

to allow delivery via a network;  (Col. 6, lines 11-16.)

-one or more files associated with the one or more extension files and describing the extension

files, the one or more files describing a logical attachment of the one or more extension files

(Col. 4, lines 21-39 and col. 6, lines 29-49.)


Mutschler failed to disclose 'software extensions being configured to allow delivery of software.

However, Voskuil disclosed:

([0014], "The client application can have the ability to be extended by external or third party

software components or modules (extensions configured for incorporation in a software

program)…a client application extension that can be transmitted to (delivery over the network)

and installed on the end user computer…", [0029], "…instructions can be incorporated in an

XML data structure that is transmitted from the server (delivery of software)…", ([0029], "The

client application supports an extension interface…allow the client application to be extended to

permit additional functionality (provide program functionality).  The extension interface permits

extension modules (e.g. plug-ins or Active-X controls) to access the file system to create, read

and write to files on the remote PC…")


Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention to modify Mutschler's invention to include a repository of software extensions to be

transmitted to a client via the network, because Mutschler's code for object models or code for

software extensions, as disclosed by Voskuil,  is merely code.  Both are stored in a repository /

database and delivered to a remote location using XML and a description of the content, in the

form of a DTD document. Both references support (Mutschler, col. 2, lines 4-5), "a need for

automatically generating format descriptions to expedite interchange..." using a common

language, (Mutschler, col. 4, lines 36-39), to "enable integration of development tools from

multiple vendors, collaboration and distribution of object-oriented design..."

**Per claim 65:**

See rejection of limitations as addressed in claim 4 above.

**Per claim 66:**

Mutschler disclosed:

-grouping multiple extension descriptions in a catalog in a network-accessible location to enable

delivery via a network; (Col. 5, lines 16-23.)

-accessing the network-accessible location; (Col. 6, lines 11-12.)

-using the catalog to update an extension that is resident on a computing device. (Col. 6, lines

22-36.)


Mutschler failed to teach software extensions and delivery of software.

However, Voskuil disclosed:

([0014], "The client application can have the ability to be extended by external or third party

software components or modules (software extensions)...a client application extension that can

be transmitted to (delivery over the network) and installed on the end user computer...", [0029],

"...instructions can be incorporated in an XML data structure that is transmitted from the server

(delivery of software)...", ([0029], "The client application supports an extension

interface...allow the client application to be extended to permit additional functionality (provide

program functionality). The extension interface permits extension modules (e.g. plug-ins or

Active-X controls) to access the file system to create, read and write to files on the remote

PC...")

Therefore, it would have been obvious, to one of ordinary skill in the art, at the time of the

invention to modify Mutschler's invention to include a repository of software extensions to be

transmitted to a client via the network, because Mutschler's code for object models or code for

software extensions, as disclosed by Voskuil, is merely code. Both are stored in a repository /

database and delivered to a remote location using XML and a description of the content, in the

form of a DTD document. Both references support (Mutschler, col. 2, lines 4-5), "a need for

automatically generating format descriptions to expedite interchange..." using a common

language, (Mutschler, col. 4, lines 36-39), to "enable integration of development tools from

multiple vendors, collaboration and distribution of object-oriented design..."

**Per claim 67:** (Col. 5, lines 21-23.)

**Per claim 68:** (Col. 5, lines 18-19.)

**Per claim 69:** (Col. 4, lines 21-23.)

### *Response to Arguments*

6.       (A) **Applicant has argued, in substance, the following:**

Applicant has pointed out (page 20, 1st paragraph), "...nowhere does Mutschler teach

software delivery over a network."

**Examiner's Response:**

Mutschler's reference has been modified to include the teachings of Voskuil. Mutschler disclosed (Abstract, lines 3-4) storing meta-model and the software systems, which store instances of the meta-model. Col. 5, lines 21-23, "The repository further includes methods for cataloging, browsing, modeling, and managing components that make up an application." (Abstract, lines 7-24), "The method comprises...extracting ...component...parsing...transformed into components of a generalized software language (XML)...distributed..." Voskuil disclosed using software extensions (plug-ins or ActiveX controls) to add functionality to a software application and to provide software delivery over a network. In both cases, code in a repository is converted to a form of XML and distributed through a networked system.

(B) Applicant **has argued, in substance, the following:**

Applicant has pointed out on page 25 that Mutschler fails to disclose 'feature types' as defined in the Specification.

**Examiner's Response:**

As can be best inferred by Examiner, 'Feature Types', refer to extensions that can be extended, creating their own tags.

(As found in Specification page 14, lines 12-19): EDFs can extend the extension mechanism...by creating their own tags...extensions can be extended...EDFs can also have one or more predefined tags...predefined XML tags for user interface elements can include tags for feature types such as tool bars, accelerators, menu items, and themes...utilized in the single navigable window application.)

This is inherent in the very definition of the XML schema, extensible mark-up language.

Structures are derived in a hierarchical manner, associated in a child / parent relationship.

A DTD states what tags and attributes are used to describe content in the XML schema.
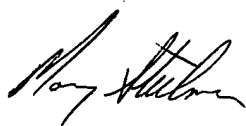
*Conclusion*

7.      The prior art made of record and not relied upon is considered pertinent to applicant's
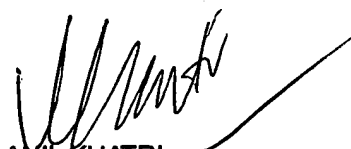
disclosure.

Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Mary Steelman, whose telephone number is (571) 272-3704. The

examiner can normally be reached Monday through Thursday, from 7:00 AM to 5:30 PM  If

attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan

Q. Dam can be reached at (571) 272-3695. The fax phone number for the organization where

this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent

Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished

applications is available through Private PAIR only. For more information about the PAIR

system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR

system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Mary Steelman

11/16/2004

ANIL KHATRI
**PRIMARY EXAMINER**